Cryptography

Lecture 11
Announcements

• HW3 due today
• HW4 is up on course webpage. Due on 3/9/20
Agenda

• Last time:
  – MACs (K/L 4.1, 4.2, 4.3)

• This time:
  – Domain Extension for MACs (K/L 4.4) and Class Exercise solutions
  – CCA security (K/L 3.7)
  – Authenticated Encryption (K/L 4.5)
Message Authentication Codes

Definition: A message authentication code (MAC) consists of three probabilistic polynomial-time algorithms \((\text{Gen, Mac, Vrfy})\) such that:

1. The key-generation algorithm \(\text{Gen}\) takes as input the security parameter \(1^n\) and outputs a key \(k\) with \(|k| \geq n\).
2. The tag-generation algorithm \(\text{Mac}\) takes as input a key \(k\) and a message \(m \in \{0,1\}^*\), and outputs a tag \(t\).
   \(t \leftarrow \text{Mac}_k(m)\).
3. The deterministic verification algorithm \(\text{Vrfy}\) takes as input a key \(k\), a message \(m\), and a tag \(t\). It outputs a bit \(b\) with \(b = 1\) meaning valid and \(b = 0\) meaning invalid.
   \(b := \text{Vrfy}_k(m, t)\).

It is required that for every \(n\), every key \(k\) output by \(\text{Gen}(1^n)\), and every \(m \in \{0,1\}^*\), it holds that \(\text{Vrfy}_k(m, \text{Mac}_k(m)) = 1\).
Unforgeability for MACs

Consider a message authentication code \( \Pi = (Gen, Mac, Vrfy) \), any adversary \( A \), and any value \( n \) for the security parameter.

Experiment \( MAC_{forge_{A,\Pi}}(n) \)

Adversary \( A(1^n) \)

\( A^{Mac_k(\cdot)} \)

\( Q \) is the set of all messages \( m' \) queried by \( A \)

Challenger

\( k \leftarrow Gen(1^n) \)

\( m' \)

\( t' \)

\( \vdots \)

\( (m, t) \)

\( MAC_{forge_{A,\Pi}}(n) = 1 \) if both of the following hold:

1. \( m \notin Q \)
2. \( Vrfy_k(m, t) = 1 \)

Otherwise, \( MAC_{forge_{A,\Pi}}(n) = 0 \)
Security of MACs

The message authentication experiment $MAC_{\text{forge}}_{A,\Pi}(n)$:

1. A key $k$ is generated by running $Gen(1^n)$.

2. The adversary $A$ is given input $1^n$ and oracle access to $Mac_k(\cdot)$. The adversary eventually outputs $(m, t)$. Let $Q$ denote the set of all queries that $A$ asked its oracle.

3. $A$ succeeds if and only if (1) $Vrf_y_k(m, t) = 1$ and (2) $m \notin Q$. In that case, the output of the experiment is defined to be 1.
Security of MACs

Definition: A message authentication code \( \Pi = (Gen, Mac, Vrfy) \) is existentially unforgeable under an adaptive chosen message attack if for all probabilistic polynomial-time adversaries \( A \), there is a negligible function \( neg \) such that:

\[
\Pr[MAC_{\text{forge}}_{A,\Pi}(n) = 1] \leq neg(n).
\]
Strong Unforgeability for MACs

Consider a message authentication code $\Pi = (Gen, Mac, Vrfy)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $MACsforge_{A,\Pi}(n)$

Adversary $A(1^n)$

$A^{Mac_k(\cdot)}$

$Q$ is the set of all message, tag pairs $(m', t')$ queried/received by $A$

Challenger

$k \leftarrow Gen(1^n)$

$MACsforge_{A,\Pi}(n) = 1$ if both of the following hold:

1. $m \notin Q$
2. $Vrfy_k(m, t) = 1$

Otherwise, $MACsforge_{A,\Pi}(n) = 0$
Strong MACs

The strong message authentication experiment $\text{MACsforg}_{A,\Pi}(n)$:

1. A key $k$ is generated by running $\text{Gen}(1^n)$.
2. The adversary $A$ is given input $1^n$ and oracle access to $\text{Mac}_k(\cdot)$. The adversary eventually outputs $(m, t)$. Let $Q$ denote the set of all pairs $(m, t)$ that $A$ asked its oracle.
3. $A$ succeeds if and only if (1) $\text{Vrf}_{\cdot_k}(m, t) = 1$ and (2) $(m, t) \notin Q$. In that case, the output of the experiment is defined to be 1.
Strong MACs

Definition: A message authentication code $\Pi = (Gen, Mac, Vrfy)$ is a strong MAC if for all probabilistic polynomial-time adversaries $A$, there is a negligible function $neg$ such that:

$$\Pr[MAC_{sforge_{A,\Pi}}(n) = 1] \leq neg(n).$$
Domain Extension for MACs
CBC-MAC

Let $F$ be a pseudorandom function, and fix a length function $\ell$. The basic CBC-MAC construction is as follows:

- **Mac**: on input a key $k \in \{0,1\}^n$ and a message $m$ of length $\ell(n) \cdot n$, do the following:
  1. Parse $m$ as $m = m_1, \ldots, m_\ell$ where each $m_i$ is of length $n$.
  2. Set $t_0 := 0^n$. Then, for $i = 1$ to $\ell$:
     - Set $t_i := F_k(t_{i-1} \oplus m_i)$.
     - Output $t_\ell$ as the tag.

- **Vrfy**: on input a key $k \in \{0,1\}^n$, a message $m$, and a tag $t$, do: If $m$ is not of length $\ell(n) \cdot n$ then output 0. Otherwise, output 1 if and only if $t = \text{Mac}_k(m)$. 
FIGURE 4.1: Basic CBC-MAC (for fixed-length messages).
Chosen Ciphertext Security
CCA Security*

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{\text{cca}}(n)$

Adversary $A(1^n)$

Challenger
CCA Security*

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{cca}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$
CCA Security*

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{\text{cca}}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$
CCA Security*

Consider a private-key encryption scheme $\Pi = (\text{Gen, Enc, Dec})$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{\text{cca}}(n)$

- **Adversary** $A(1^n)$
  - $A^{\text{Enc}_k(\cdot),\text{Dec}_k(\cdot)}$
  - $\cdot$ $m'/c'$
  - $\cdot$ $c'/m'$
  - $\cdot$ $\ldots$
  - $\cdot$ $m_0, m_1$

- **Challenger**
  - $k \leftarrow \text{Gen}(1^n)$
CCA Security*

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{cca}(n)$

Adversary $A(1^n)$

$A^{Enc_k(\cdot), Dec_k(\cdot)}$

$m'/c'$

$c'/m'$

$\vdots$

$m_0, m_1$

$c$

Challenger

$k \leftarrow Gen(1^n)$

$b \leftarrow \{0,1\}$

$c \leftarrow Enc_k(m_b)$
Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{\text{cca}}(n)$
CCA Security*

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{cca}(n)$
CCA Security* 

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary $A$, and any value $n$ for the security parameter.

Experiment $PrivK_{A,\Pi}^{\text{cca}}(n)$

$$PrivK_{A,\Pi}^{\text{cca}}(n) = 1 \text{ if } b' = b \text{ and } PrivK_{A,\Pi}^{\text{cca}}(n) = 0 \text{ if } b' \neq b.$$
CCA Security

Consider a private-key encryption scheme \( \Pi = (Gen, Enc, Dec) \), any adversary \( A \), and any value \( n \) for the security parameter. 

Experiment \( PrivK_{A,\Pi}^{cca}(n) \)

Adversary \( A(1^n) \)

\[ A^{Enc_k(\cdot), Dec_k(\cdot)} \]

\[ \begin{align*}
    m' / c' \\
    c' / m' \\
    \vdots \\
    m_0, m_1 \\
    c \\
    m' / c' \\
    c' / m' \\
    \vdots \\
    b' 
\end{align*} \]

Challenger

\[ k \leftarrow Gen(1^n) \]

\[ b \leftarrow \{0,1\} \]

\[ c \leftarrow Enc_k(m_b) \]

\[ \begin{align*}
    A^{Enc_k(\cdot), Dec^*_k(\cdot)} \\
    A \text{ may not query challenge ciphertext } c \text{ to the decryption oracle} \\
    \end{align*} \]

\[ PrivK_{A,\Pi}^{cca}(n) = 1 \text{ if } b' = b \text{ and } PrivK_{A,\Pi}^{cca}(n) = 0 \text{ if } b' \neq b. \]
CCA Security

The CCA Indistinguishability Experiment $PrivK^{cca}_{A,\Pi}(n)$:

1. A key $k$ is generated by running $Gen(1^n)$.
2. The adversary $A$ is given input $1^n$ and oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, and outputs a pair of messages $m_0, m_1$ of the same length.
3. A random bit $b \leftarrow \{0,1\}$ is chosen, and then a challenge ciphertext $c \leftarrow Enc_k(m_b)$ is computed and given to $A$.
4. The adversary $A$ continues to have oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, but is not allowed to query the latter on the challenge ciphertext itself. Eventually, $A$ outputs a bit $b'$.
5. The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.
CCA Security

A private-key encryption scheme \( \Pi = (Gen, Enc, Dec) \) has indistinguishable encryptions under a chosen-ciphertext attack if for all ppt adversaries \( A \) there exists a negligible function \( \text{negl} \) such that

\[
\Pr \left[ PrivK_{A,\Pi}^{\text{cca}}(n) = 1 \right] \leq \frac{1}{2} + \text{negl}(n),
\]

where the probability is taken over the random coins used by \( A \), as well as the random coins used in the experiment.
Authenticated Encryption

The unforgeable encryption experiment $\text{EncForge}_{A,\Pi}(n)$:

1. Run $\text{Gen}(1^n)$ to obtain key $k$.

2. The adversary $A$ is given input $1^n$ and access to an encryption oracle $\text{Enc}_k(\cdot)$. The adversary outputs a ciphertext $c$.

3. Let $m := \text{Dec}_k(c)$, and let $Q$ denote the set of all queries that $A$ asked its encryption oracle. The output of the experiment is 1 if and only if (1) $m \neq \bot$ and (2) $m \notin Q$. 
Authenticated Encryption

Definition: A private-key encryption scheme $\Pi$ is unforgeable if for all ppt adversaries $A$, there is a negligible function $neg$ such that:

$$\Pr[EncForge_{A,\Pi}(n) = 1] \leq neg(n).$$

Definition: A private-key encryption scheme is an authenticated encryption scheme if it is CCA-secure and unforgeable.